First Hit Fwd Refs

Generate Collection | Print

L3: Entry 5 of 29 File: USPT Dec 16, 2003

DOCUMENT-IDENTIFIER: US 6665746 B1
TITLE: System and method for prioritized context switching for streaming data memory transfers

Abstract Text (1):
Method of streaming data transfers from scattered locations or to a gathered location in a memory, including the steps of manipulating the transfer of data between memory devices by processing, an element at a time, entries in a first scatter/gather list; upon receipt of a request to process a second scatter/gather list, interrupting the processing of the first list; and resuming the processing of the first list upon the completion of the processing of the second. In one embodiment, the step of interrupting the processing of the first scatter gather list interrupts only at the completion of the processing of an element of the first scatter/gather list and prior to the retrieval of the next element in the first scatter/gather list. In another embodiment, the interrupting step further includes storing the next address to be processed in the first scatter/gather list, and retrieving the first element in the second scatter/gather list. Provides a scatter/gather mechanism for streaming transfers between a first and second memory device without processor or firmware intervention. The mechanism comprises: a transfer engine for streaming data transfers between dispersed locations in one memory to a gathered location in a second memory, means for suspending processing of a current transfer request upon the receipt of a second request, and means for resuming the processing of a suspended transfer request including a pointer for indicating the starting address of a next element to be processed in a suspended transfer operation. Computer program and computer program product are also provided.

Brief Summary Text (5):
The transfer of "scattered" data may be overseen by the processor. That is, the processor may be required to keep track of the scattered locations in memory associated with the related data blocks and also track the progress of the transfer. Alternatively, a scatter/gather mechanism may be employed. A scatter/gather mechanism is commonly used to stream data transfers from scattered locations or to a gathered location in a memory. A scatter/gather mechanism transfers scattered data without requiring processor (firmware) intervention. In operation, upon identifying an object (related data) for transfer, executing firmware or software creates a linked list of memory locations to be retrieved. This list (the "scatter/gather list") is transferred to a scatter/gather mechanism, which performs transfers using the list. Each individual segment in a scatter/gather list includes information about the data to be transferred and a linking address to the next address in memory for a next list segment to be processed. Individual list segments are processed by the scatter/gather mechanism sequentially until the end of the scatter/gather list is reached.

Brief Summary Text (6):
In some situations, a problem arises in the use of the scatter/gather mechanism. Typically in conventional systems and methods, once a scatter/gather mechanism begins the processing of a scatter/gather list, the resources associated with the transfer cannot be shared by others. Accordingly, transfer resources, such as a Direct Memory Access (DMA) engine associated with the scatter/gather processing,

are dedicated to that process and cannot be used or shared by any other process until the processing of a scatter/gather list has been completed. Typically, if a processor (CPU) desired to interrupt the scatter/gather mechanism, any scatter/gather operation in progress had to be aborted. This was due primarily to the lack of visibility that the processor has to the scatter/gather operation.

Brief Summary Text (10):
A method of streaming data transfers from scattered locations or to a gathered location in a memory, including the steps of manipulating the transfer of data between memory devices by processing, an element at a time, entries in a first scatter/gather list; upon receipt of a request to process a second scatter/gather list, interrupting the processing of the first scatter/gather list; and resuming the processing of the first scatter/gather list upon the completion of the processing of the second scatter/gather list. In one embodiment, this first method is further defined such that the step of interrupting the processing of the first scatter gather list interrupts only at the completion of the processing of an element of the first scatter/gather list and prior to the retrieval of the next element in the first scatter/gather list. In another embodiment, the method is further defined such that interrupting step further includes storing the next address to be processed in the first scatter/gather list, and retrieving the first element in the second scatter/gather list.

Brief Summary Text (11):
In yet another embodiment, the method is further defined such that the first scatter/gather list has a first priority, and the second scatter/gather list has a second priority; and the interrupting step is only performed if the priority of the second scatter/gather list is higher than the priority of the first scatter/gather list. This latter embodiment may also optionally provide that if the priority of the second scatter/gather list is lower than the priority of the first scatter/gather list, the method further comprises the step of delaying the processing of the second scatter/gather list until the processing of the first scatter/gather list is complete. This may be further defined to optionally provide that the step of delaying the processing of the second scatter/gather list further includes the step of storing the address in memory of the first element of the second scatter/gather list.

Brief Summary Text (12):
In another embodiment the invention provides a scatter/gather mechanism for streaming transfers between a first and second memory device without processor or firmware intervention. The the scatter/gather mechanism comprises: a transfer engine for streaming data transfers between dispersed locations in one memory device to a gathered location in a second memory device, means for suspending the processing of a current transfer request upon the receipt of a second transfer request, and means for resuming the processing of a suspended transfer request including a return pointer for indicating the starting address of a next element to be processed in an transfer operation which has been suspended.

Brief Summary Text (13):
In yet another embodiment, the invention provides a scatter/gather mechanism for streaming transfers of data between a first and second memory device without processor or firmware intervention where the data is transferred between the memory devices in segments according to instructions in a scatter/gather list, and where the scatter/gather mechanism services a single transfer request at a time. The scatter/gather mechanism includes a transfer engine for streaming transfers between dispersed locations in one memory device to a gathered location in a second memory device; and a transfer controller for processing transfer requests including a processing procedure for processing a scatter/gather list associated with a current transfer request, and an interrupt procedure for servicing a priority transfer request by suspending the current transfer request until the completion of the servicing of the priority transfer request.

Brief Summary Text (14):
In still another embodiment, the invention provides a computer system including a processor and first and second memory devices where the computer system includes: means for streaming transfers of data between the first and the second memory devices without intervention of the processor or firmware and including a scatter/gather means, where the data is transferred between the first and second memory device in segments according to instructions in a scatter/gather list, the scatter/gather means services a single transfer request at a time; and the scatter/gather means further includes a transfer engine streaming transfers between dispersed locations in the first memory device to a gathered location in a second memory device, and a transfer controller for processing transfer requests including: first processing means for processing a scatter/gather list associated with a current transfer request, and an interrupt means for servicing a priority transfer request by suspending the current transfer request until the completion of the servicing of the priority transfer request.

Brief Summary Text (15):
The invention also provides computer program, software, and firmware providing a computer program mechanism including a program module that directs the computer system or components thereof, to function in a specified manner to stream data transfers from scattered locations or to a gathered location in a memory, the program module including instructions for: manipulating the transfer of data between memory devices by processing, an element at a time, entries in a first scatter/gather list; and upon receipt of a request to process a second scatter/gather list, interrupting the processing of the first scatter/gather list; and resuming the processing of the first scatter/gather list upon the completion of the processing of the second scatter/gather list.

Detailed Description Text (2):
The present invention may be used in or in conjunction with various computing systems as a means for streaming data transfer operations. Such computing system include for example, interconnected networks of computers, distributed computing systems, stand alone computers, device controllers such as a RAID or other storage or device subsystem controller, and other information appliances. Referring to FIG. 1, an exemplary distributed computing system 50 includes a processor (CPU) 52 connected by a bus 54 to a memory controller 56, a read only memory (ROM) 58 (or other type of memory suitable for program storage, such as RAM) and an interface controller 60. Other conventional elements such as are known in the art may also be present in the computer system 50 but are not specifically described here to avoid obscuring the inventive features. For example, other input/output and user interface devices such as keyboard, pointing device, display or monitor, printers, network interface cards, hard disk or other storage devices and the like may also be present. The computer system 50 will also include an operating system and application programs. Memory controller 56 is coupled to a local memory 62. Interface controller 60 is coupled by a bus 68 to system memory 64. ROM 58 includes (stores) one or more procedures in the form of executable computer programs (a "scatter/gather list" procedure or routine) 66 for creating a scatter/gather list 72 in order to transfer dispersed data between system memory 64 and local memory 62. Interface controller 60 includes a scatter/gather mechanism 70 for processing scatter/gather lists generated by processor 52. System memory may, for example, comprise am amount of random access memory or RAM as is known in the art.

Detailed Description Text (3):
During operation, any requirement for the scattering or gathering of dispersed data results in the execution of the scatter/gather list procedure stored in ROM 58 by CPU 52. The scatter/gather list 72 is stored in local memory 62. Thereafter, CPU 52 transfers the starting address for the first list element to the scatter/gather mechanism 70 in interface controller 60 to initiate the scatter (or gather) process. The scatter/gather mechanism 70 controls the transfer of data between

local memory 62 and system memory 64.

Detailed Description Text (4):
Various types of computing systems, including distributed computing systems may benefit from operation of the inventive system and method. For example, one type of distributed computing system that can benefit from the present invention is a redundant array of independent disks (RAID) controller. A RAID controller manages the transfer of data between a host and one or more storage devices (typically hard magnetic disks) in a memory system. Memory system refers to a memory controller and the controlled memory itself. System memory refers to the memory that resides in the Host system as compared to local memory which is located in the RAID controller. In operation, the RAID controller often requires the transfer of data (in the form of either an application program or program data set) from system memory (associated with the host) to its own local memory (within the RAID controller). Here, the RAID controller may be considered a component of a larger system. Other components of such system (such as network controller) manages transfer data between host and network devices. As long as manipulation of transfer is required, the scatter and gather mechanism provided by the invention is advantageously used.

Detailed Description Text (5):
System memory is located in the system (or Host) the local memory is located in the RAID controller. Since the Interface controller that contains the DMA engine and scatter/gather sequences is located within the RAID controller, the description provided here is somewhat controller centric, thus the memory that resides on or within the RAID controller is called local memory.

Detailed Description Text (6):
Often the RAID controller data transfers require the gathering of dispersed data in the system memory 64 (e.g. memory within the system/host) for transfer to local memory 62 (e.g. memory within the RAID controller) or the scattering of data resident in local memory 62 out to dispersed locations in the system memory 64. Accordingly, a scatter/gather mechanism may be employed to perform such functions.

Detailed Description Text (9):
In operation, RAID controller 100 will typically require the transfer of data between system memory 160 and local memory 190. The transfer requested may be from a sequence of memory locations which are not contiguous in the source device, here the system memory 160. Accordingly, a scatter/gather procedure is employed. In one embodiment of the present invention, firmware resident in flash ROM 124 is executed by CPU 120 each time a scatter or gather operation needs to be performed. Typically, the gather operation requires the gathering of scattered data from system memory 160 for transfer to local memory 190. Conversely, the scatter operation requires the transfer of gathered data from local memory 190 to scattered locations in system memory 160. Alternatively, the scattering and gathering of data may be from dispersed locations in local memory 190 for transfer to contiguous locations in the system memory 160. It is noted that except where otherwise stated, scatter/gather is a commonly used technique for data transfer, and a variety of techniques are known in the art. The inventive structure and method are compatible with these known techniques and does not necessitate to use of any particular technique. The invention shows how these same scatter/gather techniques may be used with a prioritized scheme so that it can be used as context switching between multiple scatter/gather lists.

Detailed Description Text (11):
Nested element handler 306 controls the scatter/gather operation by tracking the nested return buffer in order of its priorities. The higher the priority the sooner it will it get service by the DMA engine. DMA transfer engine 170 performs data transfers between system memory 160 (See FIG. 2) and local memory 190 (See FIG. 2) by processing and executing the scatter/gather (S/G) element loaded into it. The

current scatter/gather list is processed by the DMA engine one element at a time until finish or it may get interrupted due to a higher priority scatter/gather list.

Detailed Description Text (12):
Current pointer buffer 302 stores a singular address 320 associated with the starting address 322 of the next element to be processed in a scatter gather list 324. The scatter/gather list 72 is stored in the local memory. Current pointer buffer 302 is loaded with the next address information (link address 76) each time a current list element is retrieved from local memory 190. Retrieval of the next address (link address 76) ensures the next element to be executed by the DMA is known and ready. As soon as the current element has been processed by the DMA, the next address information is used to fetch the next scatter/gather element to be executed. Retrieval is accomplished over transfer bus 129, through memory controller 130, memory bus 180, local memory 190, by interface controller 140 (See FIG. 2).

Detailed Description Text (16):
With respect to FIG. 2 and FIG. 4, local address pointer 401 indicates the starting address in local memory 190 (See FIG. 2) into which data is to be sourced (gathered) or targeted (scattered). That is, local address pointer 401 indicates the starting address in local memory 190 from which data is read if the scatter/gather operation requested requires a transfer from local memory 190 to system memory 160. Conversely, the local address pointer 401 may indicate the starting address in local memory 190 where data is to be written if the scatter/gather operation requested requires a transfer from system memory 160 to local memory 190.

Detailed Description Text (22):
Link bit 454 identifies either that there is or there is not any other element on the scatter/gather list depending upon its state. When this link bit is set (e.g. "1" there are more scatter/gather elements to be linked in the current scatter/gather list. When set, (e.g., "1") link address pointer 408 indicates the starting address in local memory for the next list element. Accordingly, the last scatter/gather list element has this bit reset (e.g., "0") and in such case, any address in link address pointer 408 is ignored.

Detailed Description Text (25):
Referring to FIG. 2, FIG. 3, and FIG. 5, a prioritized contact switching process 500 for performing nested or multiply-nested scatter/gather operations is described. By "multiply nested" we mean multiple scatter/gather lists can be processed and nested based on priority. With particular reference to the flow chart illustration of an embodiment of a method of executing nested scatter/gather operations in FIG. 5, a first scatter/gather list 72 (72-1) is generated (Step 501) by logically cascading scatter/gather elements 400 done by firmware stored in ROM 58 and CPU 120. After generation, the first scatter/gather list 72-1 is received by interface controller 140 (Step 502). Specifically, CPU 120 provides interface controller 140 with the address of the first location in local memory where the first element in the first scatter/gather list 72-1 is stored. CPU 120 has the first local memory location by virtue of the firmware running on it. In one embodiment, CPU 120 transfers the address information via local bus 125 through buffer 122. The bus arbiter (not shown) exercises control over transfer bus 129, and as appropriate allows the transfer of the address from CPU 120 to interface controller 140 in a conventional manner.

Detailed Description Text (26):
Interface controller 140 receives the address associated with the starting address in local memory 190 where the first list element in the first scatter/gather list 72-1 is stored, and stores that address (starting address in local memory) in current pointer buffer 302 (Step 504). Thereafter, nested element handler 306

enables DMA engine 170 by starting or invoking the DMA engine 170 to retrieve the first element in the first scatter/gather list 72-1 according to the local memory address stored in the current pointer buffer 302 (Step 506). The DMA transfers the list element for the first scatter/gather list. The element 400 are stored in contiguous locations in local memory. In one embodiment of the present invention, DMA engine 170 retrieves all information in each element, local address pointer, system (PCI) address pointer, byte count, configuration byte, and link address pointer.

Detailed Description Text (27):
Nested element handler 306 processes the next elements. Nested element handler 306 stores the link address associated with the starting address in memory for the next element (as indicated by link address pointer 408) in current pointer buffer 302 (Step 508). DMA engine 170 determines the type of transfer request (for example, read or write transfer request) associated with the scatter/gather list from the setting of transfer configuration byte 404 (Step 510). If a DMA read operation is indicated, as for example, during a disk write operation from the system. DMA engine 170 is enabled to perform a transfer (read) from system memory starting at the address indicated by system (PCI) address pointer 402, the number of bytes being indicated by byte count 406 from system memory 160 (Step 512). DMA engine 170 transfers the appropriate bytes from system memory 190 and writes them to local memory 160 (Step 514).

Detailed Description Text (31):
One exemplary set of rules for handling and prioritizing interrupt requests are fixed in priority to each of the interrupt buffer. For multiple interrupt buffers, each with prescribed priority, the scatter/gather list associated with each interrupt buffer will be executed according to its priority. Multiply nested processing is also readily provided by the inventive system using a plurality of return element pointers. As higher priority requests are received, the request currently being processed may be suspended or "bumped" in favor of the newly received higher priority request. The priority of the current processing is normally compared at the end of each list element, and may be suspended to resume at the next element when its priority is again determined to be the highest. The CPU with this priority knowledge can deposit the next scatter/gather list to the proper interrupt buffer (based on its priority) and expect to get the scatter/gather list serviced sooner than later with respect to the other pending scatter/gather lists.

Detailed Description Text (34):
If no other scatter/gather requests have been received, then nested element handler 306 initiates the transfer or retrieval of the next list element in the current scatter/gather list as indicated by the address stored in first buffer 302 (Step 524). Nested element handler 306 enables DMA engine 170 which retrieves the next element according to the address stored in current buffer 302. This process repeats (at step 508) where the nested element handler processes the next elements by storing the link address associated with the starting address in memory for the next element in current pointer buffer 302, until the scatter/gather list processing is complete.

Detailed Description Text (35):
If a scatter/gather list in the interrupt pointer buffer processing request has been received, then nested element handler 306 copies the link address stored in current buffer 302 (associated with the starting address of the next list element to be processed in the current scatter/gather list) to the appropriate nested return buffer 305 (Step 526). The information in the interrupt buffer will be transferred to the current buffer. Thereafter, DMA engine 170 retrieves the first list element of the scatter/gather list in the interrupt buffer. Nested element handler 306 enables DMA engine 170 to correctly retrieve the first element in the scatter/gather list of the interrupt buffer. The list element is processed

resulting in the transfer of data between local and system memory (Step 529).

CLAIMS:

1. A method of streaming data transfers from scattered locations or to a gathered location in a memory, the method comprising steps of: manipulating the transfer of data between memory devices by processing, an element at a time, entries in a first scatter/gather list; enabling interruption of said manipulating for a subset of said elements; upon receipt of a request to process a second scatter/gather list, interrupting the processing of the first scatter/gather list when interruption is enabled; and resuming the processing of the first scatter/gather list upon the completion of the processing of the second scatter/gather list.

6. The method of claim 5, wherein the step of delaying the processing of the second scatter/gather list further includes the step of storing the address in memory of the first element of the second scatter/gather list.

13. A scatter/gather mechanism for streaming transfers, which includes at least one element, between a first and second memory device without processor or firmware intervention, the scatter/gather mechanism comprising: a transfer engine for streaming data transfers between dispersed locations in one memory device to a gathered location in a second memory device; means for enabling suspension of a current transfer request for a subset of said elements; means for suspending the processing of the current transfer request upon the receipt of a second transfer request when said means for enabling is enabled; and means for resuming the processing of a suspended transfer request including a return pointer for indicating the starting address of a next element to be processed in an transfer operation which has been suspended.

14. A computer program product for use in conjunction with a computer system, the computer program product comprising a computer readable storage medium and a computer program mechanism embedded therein, the computer program mechanism, comprising: a program module that directs the computer system or components thereof, to function in a specified manner to stream data transfers from scattered locations or to a gathered location in a memory, the program module including instructions for: manipulating the transfer of data between memory devices by processing, an element at a time, entries in a fist scatter/gather list; enabling interruption of said manipulating for a subset of said elements; processing of the first scatter/gather list when interruption is enabled; and upon receipt of a request to process a second scatter/gather list, interrupting the processing of the first scatter/gather list when interruption is enabled; and resuming the processing of the first scatter/gather list upon the completion of the processing of the second scatter/gather list.

19. The computer program product of claim 18, wherein the step of delaying the processing of the second scatter/gather list further includes the step of storing the address in memory of the first element of the second scatter/gather list.

20. A computer system including a processor and first and second memory devices, and the computer system comprising: means for streaming transfers of data between the first and the second memory devices without intervention of the processor or firmware and including a scatter/gather means, where the data is transferred between the first and second memory device in segments according to instructions in a scatter/gather list, the scatter/gather means services a single transfer request at a time, the scatter/gather means further comprising: a transfer engine streaming transfers between dispersed locations in the first memory device to a gathered location in a second memory device; and a transfer controller for processing transfer requests including: first processing means for processing a scatter/gather list associated with a current transfer request, a suspension means for enabling suspension of a current transfer request for a subset of said segments, and an

interrupt means for servicing a priority transfer request when said means for enabling suspension is enabled by suspending the current transfer request until the completion of the servicing of the priority transfer request.